Proceedings of the 35th
Conference on Decision and Control
Kobe, Japan • December 1996

Avoiding Saturation By Trajectory Reparameterization[1]

George J. Pappas
Department of Electrical Engineering & Computer Sciences
University of California, Berkeley, CA 94720
gpappas@eecs.berkeley.edu

Abstract

The problem of trajectory tracking in the presence of input constraints is considered. The desired trajectory is reparameterized on a slower time scale in order to avoid input saturation. Necessary conditions that the reparameterizing function must satisfy are derived.The deviation from the nominal trajectory is minimized by formulating the problem as an optimal control problem.

1 Introduction

In this paper, the problem of trajectory tracking in the presence of input constraints is considered. The main idea of this paper is to reparameterize infeasible trajectories on a different time scale in order to satisfy the input constraints. The system will traverse the same trajectory in output space but on a slower time scale. This idea will be illustrated on feedback linearizable systems although its applicability could be much wider. Such a trajectory replanning scheme is an essential element of an automated *Air Traffic Management System* [1].

The problem of stabilization in the presence of input constraints has been considered among others in [2, 3, 4]. Stabilization results for a certain class of nonlinear systems can be found in [4], while in [5] stabilization is considered for input-output linearizable systems. In [6], model predictive methods are used to perform trajectory tracking of nonlinear systems under input constraints. An interesting idea which converts trajectory tracking to maneuver regulation was presented in [7]. In [8], reparameterization is used in order to remain within torque limitations of robotic systems.

This paper is organized as follows: In Section 2, a necessary differential equation that the reparameterizing function must satisfy for SISO systems is obtained. Minimization between the actual and the reparameterized trajectories is posed as an optimal control problem. Section 3, generalizes the concept to MIMO systems where a set of conditions that the reparameterizing function must satisfy are derived. Conclusions and issues for further research are discussed in Section 4.

[1]Research supported by the Army Research Office under grant DAAH 04-95-1-0588

0-7803-3590-2/96 $5.00 © 1996 IEEE

2 SISO Systems

The reader is assumed to be familiar with linearization results at the level of [9]. Consider the single-input, single-output nonlinear system,

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{1}$$

with state $x \in \mathbb{R}^n$, input $u \in [-M, M] \subset \mathbb{R}$, output $y \in \mathbb{R}$, vector fields f, g on $\mathbb{R}^n$ and output function h.[1] Without loss of generality, we assume that the origin is an equilibrium point of the autonomous system and that $h(0) = 0$. Let X be an open set containing the origin.

Under the condition that the nonlinear system (1) has a well defined relative degree $\gamma \leq n$ at $0 \in X$ there exists a local diffeomorphism, $\Phi : X \longrightarrow \mathbb{R}^n$, with $\Phi(0) = 0$, such that system (1) in the new coordinates is expressed as ,

$$\begin{aligned} \dot{z}_1 &= z_2 \\ &\vdots \\ \dot{z}_\gamma &= b(z,\eta) + a(z,\eta)u \\ \dot{\eta} &= q(z,\eta) \\ y &= z_1 \end{aligned} \tag{2}$$

where $z = [z_1, \ldots, z_\gamma]^T, \eta \in \mathbb{R}^{n-\gamma}$ are the zero dynamics variables, $a(z,\eta) = L_g L_f^{\gamma-1} h(x)$ and $b(z,\eta) = L_f^\gamma h(x)$ The feedback linearizing control law is

$$u = \frac{1}{a(z,\eta)}(-b(z,\eta) + v) \tag{3}$$

The new external input v is then designed to achieve reference trajectory tracking.

2.1 Exact Trajectory Tracking

From now on, we will consider the case where the relative degree of system (1) is n and therefore the system is full state, feedback linearizable. The case where $\gamma < n$ is conceptually similar and is omitted due to space restrictions. If $\gamma = n$, the state feedback law (3) becomes,

$$u = \frac{1}{a(z)}(-b(z) + v) \tag{4}$$

[1]Vector fields f,g and function h are assumed to be as smooth as needed.

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.

Let $Z = \Phi(X)$. Then $Z \subset \mathbb{R}^n$ is an open set and it contains the origin. In exact output tracking we wish that the output $y(t)$ is exactly equal to some desired trajectory $r(t) \in C^n([0, t_f])$ where one may also choose $t_f = +\infty$. Let $z_d(t) = [r(t)\ \frac{dr}{dt}(t)\ \ldots\ \frac{d^{n-1}r}{dt^{n-1}}(t)]^T$. From now on, we will assume that the desired trajectory does not exit the neighborhood X where the assumptions of relative degree hold, that is for all time $t \in [0, t_f]$ we have $z_d(t) \in Z = \Phi(X)$. Exact output tracking requires that the initial conditions satisfy $z(0) = z_d(0)$ and that the control input (3) is

$$u = \frac{1}{a(z)}(-b(z) + v) = \frac{1}{a(z_d)}(-b(z_d) + \frac{d^n r}{dt^n}) \qquad (5)$$

However, since $-M \leq u \leq M$ we must have

$$-M \leq \frac{1}{a(z_d)}\,(-b(z_d) + \frac{d^n r}{dt^n}) \leq M \qquad (6)$$

Without loss of generality we may assume that $a(z_d) > 0$. Then equation (6) implies that,

$$\frac{d^n r}{dt^n} \in [-M \cdot a(z_d) + b(z_d)\ ,\ M \cdot a(z_d) + b(z_d)] \qquad (7)$$

In order for a desired trajectory to be feasible it must satisfy necessary condition (7). A more geometric description of condition (7) may be found in [10]. Since many reference trajectories may not satisfy this necessary condition, we ask whether we can satisfy (7) by reparameterizing the desired trajectory $r(t)$ to $r(s(t))$. Note that the system still traverses the same trajectory in output space but on a different time scale.

The reparameterizing function $s(t)$ will be meaningful only if we place certain constraints on $s(t)$. First, since the desired trajectories $r(t)$ are functions belonging in $C^n([0, t_f])$ it is clear that we must have $0 \leq s(t) \leq t_f$. Second, in order to avoid going backwards in time, $s(t)$ must satisfy $\frac{ds}{dt} \geq 0$ for all t. Therefore, valid reparameterizing functions will be assumed to satisfy the following conditions:

$$0 \leq s(t) \leq t_f \qquad (8)$$

$$\tfrac{ds}{dt} \geq 0 \qquad (9)$$

Note, that $s(t)$ is allowed to be greater than t. This may result in acausal control laws where the system looks ahead at the trajectory and tries to exploit situations where input saturation does not occur. If acausal control laws are not desired, then $s(t)$ must satisfy $0 \leq s(t) \leq t$.

We now derive necessary conditions that the reparameterizing function $s(t)$ must satisfy. Consider the reparameterized trajectory $r(s(t))$. Simple calculations show that,

$$\frac{d^n r}{dt^n} = T(s, \frac{ds}{dt}, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) + r^{(1)}\frac{d^n s}{dt^n} \qquad (10)$$

where the nature of the nonlinear function T depends on the relative degree of the system as well as the desired trajectory. Plugging equation (10) into equation (7) we obtain,

$$r^{(1)}\frac{d^n s}{dt^n} \in [S_1, S_2] \qquad (11)$$

where $S_1 = -M \cdot a(z_d) + b(z_d) - T$ and $S_2 = M \cdot a(z_d) + b(z_d) - T$. Recall that $z_d(t) = [r(t)\ \frac{dr}{dt}(t)\ \ldots\ \frac{d^{n-1}r}{dt^{n-1}}(t)]^T$. From equation (10) it is clear that $z_d(s(t))$ is a function of $s, \frac{ds}{dt}, \ldots, \frac{d^{n-1}s}{dt^{n-1}}$. Defining

$$\begin{aligned} A(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) &= a(z_d(s(t))) \\ B(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) &= b(z_d(s(t))) \\ S_1(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) &= -M \cdot A + B - T \\ S_2(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) &= M \cdot A + B - T \end{aligned}$$

results in the following necessary condition,

$$r^{(1)}\frac{d^n s}{dt^n} \in [S_1(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}})\ ,\ S_2(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}})] \qquad (12)$$

Therefore given a feedback linearizable system, input bounds and a desired output trajectory $r(t)$, one chooses the reparameterizing function $s(t)$ so that necessary condition (12) is satisfied. We now consider two cases:

Case I : $r^{(1)} \neq 0$ Without loss of generality we can assume that $r^{(1)} > 0$. In this case, condition (12) can be written as

$$\frac{d^n s}{dt^n} \in [\frac{1}{r^{(1)}}S_1\ ,\ \frac{1}{r^{(1)}}S_2] \qquad (13)$$

If we define $s_1 = s, \ldots, s_n = \frac{d^{n-1}s}{dt^{n-1}}$, then necessary condition (13) requires that the reparameterizing function $s(t)$ satisfies the following controlled differential equation

$$\begin{aligned} \dot{s}_1 &= s_2 \\ &\vdots \\ \dot{s}_{n-1} &= s_n \\ \dot{s}_n &= w \end{aligned} \qquad (14)$$

where the control w must take values in the interval W defined as

$$W = [\frac{1}{r^{(1)}}S_1(s_1, \ldots, s_n)\ ,\ \frac{1}{r^{(1)}}S_2(s_1, \ldots, s_n)] \qquad (15)$$

In order to ensure that reparameterization does not affect the differentiability of the trajectory, the initial conditions are chosen so that $z_d(0) = z_d(s(0))$. From equation (10) we obtain

$$\begin{aligned} s_1(0) &= s(0) = 0 \\ s_2(0) &= \frac{ds}{dt}(0) = 1 \\ s_3(0) &= \frac{d^2 s}{dt^2}(0) = 0 \\ &\vdots \\ s_n(0) &= \frac{d^{n-1}s}{dt^{n-1}}(0) = 0 \end{aligned} \qquad (16)$$

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.

which satisfy conditions (8,9). If $z_d(0) = z_d(s(0))$ is not required, then one could choose any initial conditions which satisfy (8,9).

Note that $s(t)$ satisfies a controlled differential equation which implies that there may be many solutions for $s(t)$ which avoid input saturation. However, the question now becomes whether the system will reach its final goal, namely whether we can steer s_1 to the desired value t_f. Furthermore, any valid solution to the above problem must also satisfy conditions (8,9). But condition (9) along with $s_1(0) = 0$ imply that condition (8) will be satisfied since once $s_1 = t_f$, the problem is solved. Hence, existence of a valid reparameterization function can be shown by solving the following problem.

Problem 1 *Given the controlled differential equation (14), initial conditions (16) and the control interval W defined in (15), choose the control $w \in W$ so as to steer s_1 to t_f while satisfying the constraint $s_2 \geq 0$.*

In case there are many solutions which satisfy the above constraints, we choose the reparameterizing function $s(t)$ which steers s_1 to t_f in minimum time.

Problem 2 *Given the controlled differential equation (14), initial conditions (16) and the control interval W defined in (15), choose the control $w \in W$ so as to steer s_1 to t_f in minimum time while satisfying the constraint $s_2 \geq 0$.*

Note that in the above optimal control problem, the system is linear, the cost is convex and the search space is convex. The final time is free, but the final state is constrained. The above control problem is difficult because of the state dependent control interval W as well as the state constraint. The nonlinearities of the system (1) and the desired trajectory $r(t)$ are all hidden in the control interval W. Because of the choice of optimization cost and since W is a bounded and state dependent interval, solutions to the above optimal control problem will be of bang-bang nature.

Case II: $r^{(1)} = 0$ In the degenerate case where the first derivative of the reference trajectory vanishes, necessary condition (12) becomes

$$0 \in [S_1(s_1, \ldots, s_n)\ ,\ S_2(s_1, \ldots, s_n)] \qquad (17)$$

which implies that $\frac{d^n s}{dt^n}$ is unconstrained as long as condition (17) is satisfied. If condition (17) is not satisfied when $r^{(1)} = 0$ then saturation cannot be avoided by reparameterization. Note that in this case the idea of reparameterization is meaningless since the desired trajectory freezes and thus reparameterization serves no purpose. Two cases must be considered. The first one occurs when $r^{(1)}(s) = 0$ at some particular point s, and the second when $r^{(1)}(s) \equiv 0$ for all s in some interval S which has nonzero measure.

In the first case where $r^{(1)}(s) = 0$ at some particular point s it is assumed that condition (17) is satisfied for $s_1 = s$. But then, by smoothness of solutions, we obtain that (17) is satisfied for $s_1 = s_-$ as well as for $s_1 = s_+$. Furthermore $r^{(1)}(s_+) \neq 0$ and we can therefore proceed by solving again the optimal control problem 2 after s_+. We can therefore jump over points where the reference trajectory freezes momentarily as long as condition (17) is satisfied. If condition (17) is satisfied then the control input w in the differential equation (14) is essentially unconstrained and we could choose $w = w_-$ in order to preserve continuity of the control input w.

We now consider the case where $r^{(1)}(s) \equiv 0$ for all $s(t) \in [\sigma_1, \sigma_2]$ or equivalently for all $t \in [t_1, t_2]$. If $r^{(1)}(s) = 0$ for all $s(t) \in [\sigma_1, \sigma_2]$, then by equations (10) it is clear that

$$\begin{aligned} r(s(t)) &= r(t) \text{ for } t \in [t_1, t_2] \\ \frac{dr}{dt} &= r^{(1)} \frac{ds}{dt} = 0 \\ &\vdots \\ \frac{d^n r}{dt^n} &= T(s, \frac{ds}{dt}, \ldots, \frac{d^{n-1}s}{dt^{n-1}}) + r^{(1)} \frac{d^n s}{dt^n} = 0 \end{aligned}$$

and therefore the state will be at $[r(s), 0, \ldots, 0]^T$ until $s = \sigma_2$ or equivalently $t = t_2$. The nominal trajectory $r(t)$ may be also stationary (if $t < t_2$) or may be non-stationary (if $t \geq t_2$). We can take advantage of the fact that the trajectory is frozen for some time by reseting the reparameterizing function $s(t)$ in the following way

$$s(t) = \begin{cases} t & \text{if } t < t_2 \\ t_2 & \text{if } t \geq t_2 \end{cases} \qquad (18)$$

Therefore if $t < t_2$, then the nominal and the reparameterized trajectories are both frozen at the same location in state space. Then we can simply set $s(t)$ equal to t in order to catch up to the nominal time parameterization. If, however, $t \geq t_2$ then the nominal trajectory $r(t)$ is no longer stationary and therefore the best the reparameterized trajectory can do is catch up to the last point where $r(t)$ was stationary. Therefore, in this case we set $s(t) = t_2$. When $t > t_2$, then $r^{(1)} \neq 0$ and we can proceed by solving again the optimal control Problem 2.

2.2 Asymptotic Tracking

We consider using reparameterization to avoid saturation in asymptotically tracking a reference trajectory. In case the tracking error is large, reparameterization will slow down the desired trajectory and distribute more control authority to stabilizing the tracking error. Note, however, that since the feedback law for trajectory tracking using feedback linearization contains a linear function of the error, sufficiently large tracking errors will saturate the control and, in that case, the trajectory needs to be replaned.

In asymptotic tracking we have $z(0) \neq z_d(0)$. The following change of coordinates

$$\begin{aligned} e_1 &= z_1 - r \\ &\vdots \\ e_n &= z_n - \frac{d^n r}{dt^n} \end{aligned}$$

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.

or more compactly $e = z - z_d$ results in

$$\begin{aligned} \dot{e}_1 &= e_2 \\ &\vdots \\ \dot{e}_{n-1} &= e_n \\ \dot{e}_n &= b(z) + a(z)u - \frac{d^n r}{dt^n} \end{aligned} \tag{19}$$

If $s^n + c_{n-1}s^{n-1} + \ldots + c_1 s + c_0$ is a Hurwitz polynomial then the following control law

$$u = \frac{1}{a(z)}(-b(z) - c^T e + \frac{d^n r}{dt^n}) \tag{20}$$

will ensure that the error e converges to zero exponentially. However, since $-M \leq u \leq M$ we must ensure that the input bounds are not violated. Since $z = z_d + e$, we must have

$$-M \leq \frac{1}{a(z_d+e)}(-b(z_d+e) - c^T e + \frac{d^n r}{dt^n}) \leq M$$

Notice that we must ensure that $z \in \Phi(X)$. This may potentially place restrictions on how large the initial error e can be. For sufficient conditions on the initial error which guarantee that $z \in \Phi(X)$ see [10].

We will again use a reperameterization algorithm in order to avoid saturation. The only difference in this case is the addition of the tracking error. Using equation (10), define

$$\begin{aligned} A(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}, e) &= a(z_d + e) \\ B(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}, e) &= b(z_d + e) \\ S_1(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}, e) &= -M \cdot A + B + c^T e - T \\ S_2(s, \ldots, \frac{d^{n-1}s}{dt^{n-1}}, e) &= M \cdot A + B + c^T e - T \end{aligned}$$

and assuming without loss of generality that $a(z_d+e) > 0$, results in the following necessary condition,

$$r^{(1)}\frac{d^n s}{dt^n} \in [S_1 \ , \ S_2] \tag{21}$$

As in the exact tracking case, we can consider cases where the trajectory becomes stationary or not. Problems similar to Problems 1 and 2 can be formulated by simply appending the error dynamics to the formulation. Although the problems are more difficult in this case, they are conceptually the same. The main difference is the dependence of the reparameterization function on the tracking error. This allows the system to reparameterize the trajectory depending on how large the tracking error is. Therefore, if the tracking error is large then the system distributes more control authority towards trajectory tracking by slowing down the trajectory. Once the tracking error converges to zero, the above results reduce to the exact tracking results. The system will then be able to distribute more control authority towards speeding up the trajectory in order to catch up with the nominal one.

3 Extension to MIMO Systems

We would like to extend the reparameterization story obtained in the last section to MIMO nonlinear systems. Consider the square MIMO nonlinear control system described by

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x)u_i \\ y_j &= h_j(x) \quad j = 1, \cdots, m \end{aligned} \tag{22}$$

where $x \in \mathbb{R}^n$, $u_i \in [-M_i, M_i] \subset \mathbb{R}$, $f(x)$, $g_1(x), \cdots, g_m(x)$ are smooth vector fields on $\mathbb{R}^n$ and $h_1(x), \cdots, h_m(x)$ smooth functions on $\mathbb{R}^n$. Let x_0 be an equilibrium point of the autonomous system, that is $f(x_0) = 0$, and let $h_j(x_0) = 0$ for all $j = 1, \ldots, m$.

We *assume* in what follows that each output y_j has a well defined *relative degree* γ_j, in an open neighborhood X of x_0, and the decoupling matrix $D(x)$ (where $D_{ij}(x) = L_{g_j}L_f^{\gamma_i - 1}h_j(x)$) is nonsingular in X. Under these assumptions, there exists a diffeomorphism $(z, \eta) = \Phi(x)$ such that the state feedback given by

$$u = (D(x))^{-1}\ [\ -d(x) + v\ ] \tag{23}$$

where $d_i(x) = L_f^{\gamma_i}h_i(x)$, will result in a closed loop system which is decoupled from the new input v to output y. For simplicity, we consider from now on the case where the MIMO system is full state feedback linearizable. The presence of internal dynamics does not add any conceptual complications in the following formulation and is omitted due to space limitations.

3.1 MIMO Exact Tracking

In order for the outputs y_i to exactly track the reference trajectories $r_i(t)$, the external inputs v_i are chosen as

$$\begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} = \begin{bmatrix} \frac{d^{\gamma_1} r_1}{dt^{\gamma_1}} \\ \vdots \\ \frac{d^{\gamma_m} r_m}{dt^{\gamma_m}} \end{bmatrix} \tag{24}$$

If we define $\Delta(x) = \delta_{ij}(x) = (D(x))^{-1}$ as the inverse of the decoupling matrix and the vector $\beta(x)$ as

$$\begin{bmatrix} \beta_1(x) \\ \vdots \\ \beta_m(x) \end{bmatrix} = \Delta(x) \cdot d(x) = \Delta(x) \cdot \begin{bmatrix} d_1(x) \\ \vdots \\ d_m(x) \end{bmatrix} \tag{25}$$

then the feedback linearizing control law (23) has to satisfy

$$|\beta_i(x) + \Delta(x)v_i| \leq M_i \tag{26}$$

for $1 \leq i \leq m$. Note that we have transformed the MIMO problem into m decoupled SISO problems. This was done by implicitly assuming that $\Delta(x)$ is known either analytically or numerically. Combining equations (24,26) results in the following set of necessary conditions for the reference trajectories,

$$-M_i \leq \beta_i(x) + \sum_{j=1}^{m} \delta_{ij}(x)\frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \leq M_i \tag{27}$$

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.

or otherwise,

$$\sum_{j=1}^{m} \delta_{ij}(x) \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \in [-M_i - \beta_i(x), M_i - \beta_i(x)] \quad (28)$$

Condition (28) is a necessary condition that must be satisfied by all output reference trajectories r_i. If it is not satisfied for any i, we again ask ourselves whether we could satisfy the above condition by reparameterizing time to $s(t)$. Note that in order to traverse the same trajectory in output space, the reparameterizing function $s(t)$ must be the *same* for all outputs r_i.

Recall that by reparameterizing $r_i(t)$ to $r_i(s(t))$, the time derivatives of the outputs are of the form,

$$\frac{d^k r_i}{dt^k} = T_i(s, \frac{ds}{dt}, \ldots, \frac{d^{k-1}s}{dt^{k-1}}) + r_i^{(1)} \cdot \frac{d^k s}{dt^k} \quad (29)$$

Since $x = \Phi^{-1}(z)$ and state z is a function of the reference trajectories and their derivatives, by equation (29) it is clear that the state x depends on $s, \frac{ds}{dt}, \ldots, \frac{d^{\gamma_i} s}{dt^{\gamma_i}}$ for $1 \leq i \leq m$. Thus, necessary conditions (28) become

$$\sum_{j=1}^{m} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \in [-M_i - \beta_i, M_i - \beta_i] \quad (30)$$

where now every term in condition (30) is a function of s, $\frac{ds}{dt}$, ..., $\frac{d^{\gamma_i} s}{dt^{\gamma_i}}$ for $1 \leq i \leq m$. In the SISO case, the equivalent condition is condition (7). Note that in the SISO case, the highest derivative of the reference trajectory must satisfy condition (7). For MIMO systems, because of the decoupling matrix, it is no longer clear how to extract the highest derivative from conditions (30) since many of the δ_{ij} may be zero and furthermore many of the outputs may have the same relative degree.

Consider condition (30) for a fixed $i \in \{1, \ldots, m\}$. Let γ_i^* be the largest derivative degree appearing in (30). Define the set index K_i as

$$K_i = \{1 \leq j \leq m \mid \gamma_j = \gamma_i^* \text{ and } \delta_{ij} \neq 0\} \quad (31)$$

We can then break up the following sum

$$\begin{aligned} \sum_{j=1}^{m} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} &= \sum_{j \in K_i} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} + \sum_{j \notin K_i} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \\ &= \left(\sum_{j \in K_i} \delta_{ij} \right) \frac{d^{\gamma_i^*} r_j}{dt^{\gamma_i^*}} + \sum_{j \notin K_i} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \end{aligned} \quad (32)$$

Using equation (29) we can furthermore obtain

$$\begin{aligned} \sum_{j=1}^{m} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} &= \left(\sum_{j \in K_i} \delta_{ij} \right) \frac{d^{\gamma_i^*} r_j}{dt^{\gamma_i^*}} + \sum_{j \notin K_i} \delta_{ij} \frac{d^{\gamma_j} r_j}{dt^{\gamma_j}} \\ = \left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right) \frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} &+ \sum_{j \notin K_i} \delta_{ij} r_j^{(1)} \frac{d^{\gamma_j} s}{dt^{\gamma_j}} + \sum_{j=1}^{m} \delta_{ij} T_j \end{aligned}$$

If we define the function Γ_i as

$$\Gamma_i = \sum_{j \notin K_i} \delta_{ij} r_j^{(1)} \frac{d^{\gamma_j} s}{dt^{\gamma_j}} + \sum_{j=1}^{m} \delta_{ij} T_j(s, \ldots, \frac{d^{\gamma_j - 1} s}{dt^{\gamma_j - 1}}) \quad (33)$$

then necessary condition (30) can be written as

$$\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right) \frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} \in [-M_i - \beta_i - \Gamma_i, M_i - \beta_i - \Gamma_i]$$

Note that the above necessary condition must be satisfied for each $1 \leq i \leq m$. If $\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right) = 0$, then it is clear from the above condition that the trajectory must satisfy the condition

$$0 \in [-M_i - \beta_i - \Gamma_i, M_i - \beta_i - \Gamma_i] \quad (34)$$

Therefore, if condition (34) is satisfied then $\frac{d^{\gamma^*} s}{dt^{\gamma^*}}$ is unconstrained and thus $\frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} \in \mathbb{R}$ If condition (34) is not satisfied then the necessary condition is not satisfied and thus $\frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} \in \emptyset$. If $\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right) \neq 0$ then we can assume without loss of generality that $\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right) > 0$ and therefore the above condition can be equivalently expressed as

$$\frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} \in S_i \quad (35)$$

where the interval S_i is defined as

$$S_i = [\frac{(-M_i - \beta_i - \Gamma_i)}{\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right)}, \frac{(M_i - \beta_i - \Gamma_i)}{\left(\sum_{j \in K_i} \delta_{ij} r_j^{(1)} \right)}] \quad (36)$$

Grouping the above we can see that for each $1 \leq i \leq m$ we have that

$$\frac{d^{\gamma_i^*} s}{dt^{\gamma_i^*}} \in W_i = \begin{cases} S_i & \text{if } \sum_{j \in K_i} \delta_{ij} r_j^{(1)} \neq 0 \\ \mathbb{R} & \text{if } \sum_{j \in K_i} \delta_{ij} r_j^{(1)} = 0 \\ & \text{and (34) holds} \\ \emptyset & \text{otherwise} \end{cases} \quad (37)$$

It might be the case, however, that for $i_1 \neq i_2$ we have $\gamma_{i_1}^* = \gamma_{i_2}^*$. In that case condition (37) results in two constraints on $\gamma_{i_1}^*$-th derivative of $s(t)$. In that case we must have

$$\frac{d^{\gamma_{i_1}^*} s}{dt^{\gamma_{i_1}^*}} = \frac{d^{\gamma_{i_2}^*} s}{dt^{\gamma_{i_2}^*}} \in W_{i_1} \bigcap W_{i_2}$$

Motivated by the above we define the index sets

$$N_k = \{1 \leq j \leq m \mid \gamma_j^* = k\} \quad (38)$$

The index sets N_k are essentially a partition of the index set $\{1, \ldots, m\}$ under the following equivalence relation: indices i_1 and i_2 are equivalent if $\gamma_{i_1}^* = \gamma_{i_2}^*$. We can then group conditions (37) and thus the k-th derivative of $s(t)$ must satisfy the following set of constraints

$$\frac{d^k s}{dt^k} \in V_k = \bigcap_{i \in N_k} W_i \quad (39)$$

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.

Conditions (39) are the conditions that the reparameterizing function must satisfy. Note that the problem has no solution if the set V_k for any k is the empty set. This may happen in two ways. First, any of the W_i may be empty. This means that reparameterization cannot avoid saturation in the input-output channel i. Second, the intersection of the W_i may be empty which means that it is impossible to avoid saturation in all input channels simultaneously by reparameterization.

Let k^* be the highest number k for which N_k is nonempty. In other words, k^* is the degree of the highest derivative of $s(t)$ that is constrained in conditions (39). If we define $s_1 = s, \ldots, s_{k^*} = \frac{d^{k^*-1}s}{dt^{k^*-1}}$, then the reparameterizing function $s(t)$ must satisfy the following controlled differential equation

$$\begin{aligned} \dot{s}_1 &= s_2 \\ &\vdots \\ \dot{s}_{k^*} &= w \end{aligned} \tag{40}$$

as well as the state constraints

$$\begin{aligned} s_2 &\in V_1 \\ &\vdots \\ s_{k^*} &\in V_{k-1} \end{aligned} \tag{41}$$

in addition to the previous state constraint (9)

$$s_2 \geq 0 \tag{42}$$

and the control must be in $w \in V_k$. Existence of a reparameterizing solution can thus be shown by solving the following problem.

Problem 3 *Given the controlled differential equation (40), choose the control $w \in V_k$ in order to steer s_1 to t_f while satisfying state constraints (41,42).*

In case of nonunique solution, redundancy is resolved by solving the following optimal control problem.

Problem 4 *Given the controlled differential equation (40), choose the control $w \in V_k$ in order to steer s_1 to t_f in minimum time while satisfying state constraints (41,42).*

Note that the initial conditions must also satisfy state constraints (41,42). This may result in initial conditions different from (16), which could result in loss of smoothness. In that case, however, the initial conditions can be treated as parameters in which case the optimization can be performed over the initial conditions as well. The above optimal control problem may be very difficult to solve, even numerically, since we are trying to satisfy many constraints by varying only one parameter. The complexity of the above formulation should be expected since we want to avoid saturation in many input output channels by altering one parameter.

4 Conclusions

The problem of reparameterizing desired output trajectories in order to avoid input saturation was considered. Necessary conditions that the reparameterizing function must satisfy were derived for SISO and MIMO feedback linearizable systems. The deviation from the nominal trajectory was minimized by formulating an optimal control problem. The formulation of the optimal control problem raises several issues such as existence of solutions, structure of optimal control laws etc. Necessary conditions for other classes of systems and feedback control laws will be investigated in order to show the generality of the main idea.

Acknowledgment: The author would like to thank John Lygeros, Datta Godbole and John-Morten Godhavn for many interesting discussions on this subject as well as his advisor, Prof. S.S. Sastry, for being a continuous source of inspiration.

References

[1] S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole, and G. Pappas. Hybrid control in air traffic management systems. In *Proceedings of the 1995 IEEE Conference in Decision and Control*, pages 1478–83, New Orleans, LA, December 1995.

[2] Z. Lin and A. Saberi. Semi-global exponential stabilization of linear systems subject to input saturation via linear feedbacks. *Systems & Control Letters*, 21(3):225–239, 1993.

[3] E.D. Sontag and H.J. Sussmann. Nonlinear output feedback design for linear systems with saturating controls. In *Proceedings of the 29th IEEE Conference on Decision and Control*, pages 3414–3416, 1990.

[4] A. R. Teel. *Feedback Stabilization: Nonlinear Solutions to Inherently Nonlinear Problems.* PhD thesis, Department of Electrical Engineering, University of California, Berkeley, California, 1992.

[5] A.R. Teel and L. Praly. Tools for semi-global stabilization by partial state and output feedback. *SIAM Journal of Control and Optimization*, 33(5):1443–88, September 1995.

[6] P. Lu. Constrained tracking control of nonlinear systems. *Systems & Control Letters*, 27(5):305–314, 1996.

[7] J. Hauser and R. Hindman. Maneuver regulation from trajectory tracking: Feedback linearizable systems. In *Proceedings of NOLCOS '95*, Lake Tahoe, CA, June 1995.

[8] O. Dahl and L. Nielsen. Torque-limited path following by online trajectory time scaling. *IEEE Transactions on Robotics and Automation*, 6(5):554–561, 1990.

[9] A. Isidori. *Nonlinear Control Systems.* Communications and Control Engineering Series. Springer Verlag, New York, 1989.

[10] G.J. Pappas, J. Lygeros, and D.N. Godbole. Stabilization and tracking of feedback linearizable systems under input constraints. In *Proceedings of the 34th IEEE Conference on Decision and Control*, pages 596–601, New Orleans, LA, December 1995.

Authorized licensed use limited to: University of Pennsylvania. Downloaded on October 29,2026 at 15:32:27 UTC from IEEE Xplore. Restrictions apply.